Literature generating using Markov Chains

Project Report

Mohammad-Ali Balaj

If you require any further information, feel free to contact me.
Markov Chains Projects GitHub-Repository.

Field: Engineering & IT
Date: January 18, 2023

Andrey Andreyevich Markov (14 June 1856 – 20 July 1922) was a Russian mathematician best known for his work on stochastic processes. A primary subject of his research later became known as the Markov chain [1].

Contents

List of Figures

1 Abstract

Data scientists face uncertainty sooner or later in their research journey. For this reason, a precise estimation of probability is crucial. One of the excellent ways to do it, is using Markov Chains which is the most useful and popular class of stochastic processes in real-world applications. It is a mathematical concept, which is defined as a set of random variables, that transform from one state to another state, based on certain probabilistic rules. In this project the focus is on generating various literature such as meaningful sentences, stories and lyrics.

2 Introduction

Markov Chain is a simple concept which is used in many real-world applications. It models time and space-dependent stochastic processes and is based on a simple rule namely, everything that will happen in the future only depends on what is happening right now! Although the concept is straightforward, it can model most complicated real-time processes which is used in different domains like finance, sales, NLP algorithms, music composition, speech recognition, weather forecasting, Google's page rank algorithms to make their predictions easily and accurately. It is developed by Andrey Markov in 1906.

3 Basic Concepts

In order to define Markov Chain we need a brief review of some basic but important concepts in probability theory.

3.1 Stochastic Process

Stochastic process is a process of some values changing randomly over the time. Although stochastic literally means random but they can be entirely deterministic. It is a mathematical process that can be modeled with a family of random variables. There are different types of stochastic processes, such as famous Markov chains, random walks and Bernoulli processes. There are 2 different types of stochastic processes namely, discrete-time and continuous. For instance, in the classical tossing coin example, the probability of the coin landing on heads is 0.5, and tails also 0.5. If the coins lands on heads, the 'state' is unchanged and if the coin lands on tails, the 'state' has changed. This can be modeled with the graph below (T stands for tails and H stands for heads):

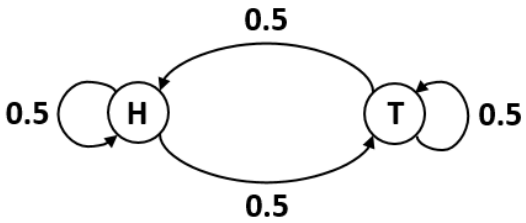

Figure 1: Probability graph of changing states (tossing a coin)

3.2 Random Variables

A random or a stochastic variable is a variable whose value is unknown. Also it could be a function that helps determine an event's probability by assigning a quantity to the outcome. These variables can be discrete or continuous. The following set below is a sequences of random variables X indexed by some time parameter t, which can possess the Markov property.

$$X_0, X_1, X_2 \ldots, X_t, \ldots \tag{1}$$

3.3 Markov Property

Markov property refers to a stochastic process which is memoryless. It simply means that the probability of the next state only depends on the current state, everything before the current state is irrelevant. As shown below the Markov property is described mathematically.

$$P(X_{t+1} = x | X_1 = x_1, X_2 = x_2, \ldots, X_t = x_t) = P(X_{t+1} = x | X_t = x_t) \quad (2)$$

4 Markov Chains

The concept of Markov property which is explained in previous section, is used to construct a basic Markov Chain. It is simply a sequence of random variables that take on states in the given state space. For instance if we toss the coin 5 times, the observed sequence could be Heads, Tails, Tails, Heads and Heads which is represented as a sequence of random variables mathematically. Each state X_i in the following sequence is a random variable which has a particular probability of occurring, described by transition probabilities between each state.

$$X_0, X_1, X_2, X_3, X_4 \tag{3}$$

The following sequence is the corresponding transition probabilities of between the sequence of changing states as described above:

$$P_{0,1}, P_{1,2}, \ldots, P_{(t-1),t} \tag{4}$$

The following graph indicates the changing of states also called transitions, with a passage of time according to their probabilities.

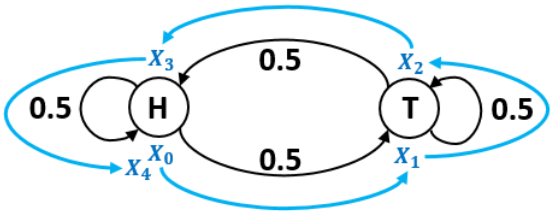

Figure 2: Sequence of random variables (tossing a coin)

5 Text generation using Markov Chains

As known the aim of this project, is to generate various texts using Markov Chain. After getting familiar with basic concepts in previous sections, the focus is on developing some text generator applications using Markov Chains. Due to their useful properties, they are also used in various fields such as statistics, biology/medicine, modelling of biological populations evolution, computer science, information theory and speech recognition and many others. In addition to that some NLP techniques are applied in order to process the sample data. Natural Language Processing or NLP for short, is a branch of artificial intelligence that is growing very fast in various fields such as research, business, IT and many more. The ultimate objective of NLP is to understand human languages, interpret them and gaining insights from them. There are many real-world problems, which could be solved using NLP. For instance detection of spam emails, summarizing a text, sentiment analysis, topic modeling and the famous text generation.
In one of the six implemented applications, a Recurrent Neural Network or RNN (LSTM architecture) is used which is a type of artificial neural network to detect the sample patterns and produce more realistic lyrics.

6 Lyrics / Poetry Generation With Markov Chains

6.1 Metadata

The Song Lyrics dataset is published by Paul Mooney in Kaggle. It is an open-source dataset (license: CC0: Public Domain) and contains 49 lyrics in text format.

6.2 Implementation

A Recurrent Neural Networks (LSTM function) from Keras in addition to Markov chain (Markovify Python library) is used in order to implement this project. The goal of this project is to generate new verses in the style of the poems, that is given as an input which predict the properties of the next poem line. This project is inspired by Peter Potash papers [2, 3]. After installing and importing needed Python libraries, the lyrics data from my favourite artists are explored. For instance the bar charts shown below indicate the number of words in Notorious B.I.G. and Lil Wayne lyrics.

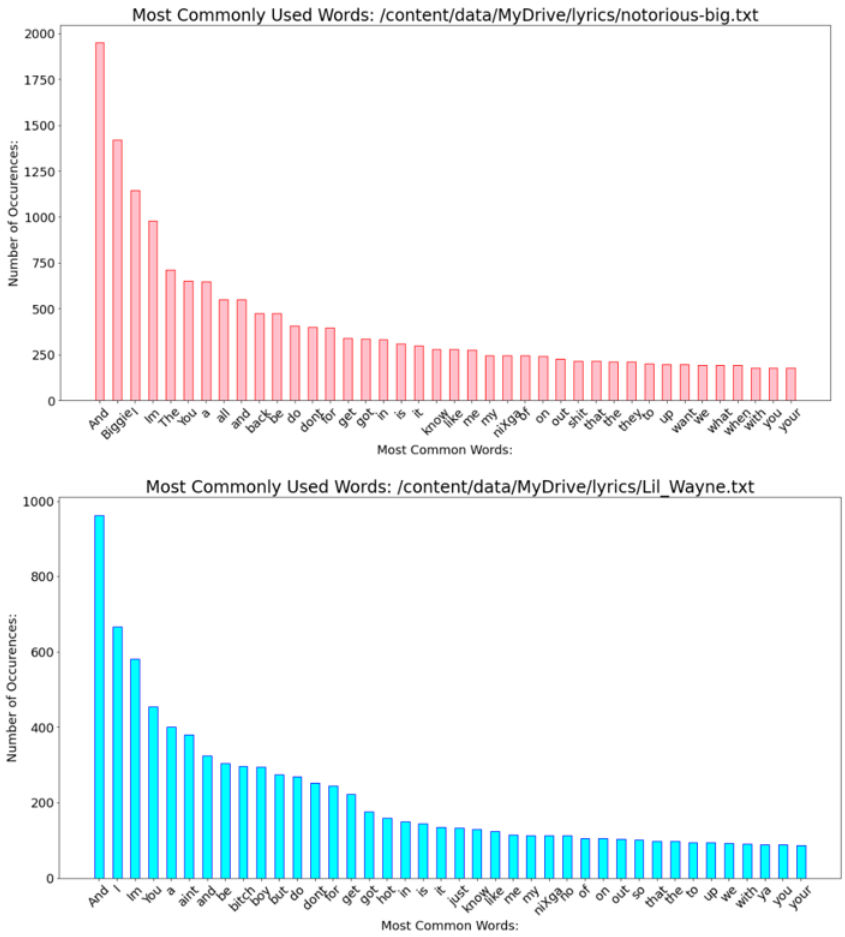

Figure 3: Notorious B.I.G. and Lil Wayne lyrics descriptive statistics

Following steps are performed in order to generate new verses:

- Design the Recurrent Neural Network
- Defining Markov Chain model
- Determine number of syllables in line
- Indexing of words rhyme
- Making rhymes
- Separate each line of the input text
- Defining the lyrics generator function
- Building the dataset
- Composing verses
- Training Phase and running the model with adjusted parameters and hyper-parameters

The following comparison of original lyrics of Lil Wayne and the generated one, demonstrates our model's performance. For more details and complete results please check the project GitHub repository.

Original lyrics of Lil Wayne:

```
They call me Mr Carter I kissed the daughter
Of the deads forehead I killed the father
Spilled the heart of a mildew hater
I will put them body on chill like glaciers
Gracias Im crazy yes its obvious
Make harm in the bomb you in public
```

Generated lyrics in Lil Wayne style:

```
I dont like me I am the wildest
Put ya in the heat get out limp
Cause thats what you doin mam
Then what you like them I coudlnt
And this the way they said i couldnt
So high in the mirror on the whip
```

7 Book Generation With Markov Chains

7.1 Metadata

The Books in plain text dataset, is published by Data Storm in Kaggle. It is an open-source dataset (license: CC0: Public Domain) and contains about 18000 books in text format. A subset of this dataset is used in this project.

7.2 Implementation

After installing and importing Natural Language Toolkit and needed Python libraries, reading and text cleaning process is performed on each book. As next step the Markov function is implemented including the calculation of transition probabilities for 10189936 states in this case. Finally the cleaned text is fitted to the model in order to generate books. A reading sample of first generated book is shown below. For more details and complete results please check the project GitHub repository.

A reading sample of first generated book:

```
because i was partly both i loved whathe did not hear
him as  a darling boy but how can he make a prior
drug convictionyou are gods gift the door one set
was in an indian that escaped the hair removing
cream she rubbed my back mandy was telling the truth
about leaving i offer them a steady rhythm rick was
still a thrill of excitement shed been struck unable
to speak english all transliterations of greek words
genos and genea the first refers to one only passion
she said it was nothing fine i sighed and carried on
hernandez or me he just broke his neck she turned and
went home full and she was just about to take any more
words he learned the part that was because there was a
shooting at a normal pace sharlene stands up to her
cheekeverything settled back down harvey and mo are you
busy not at all cheiron the lord by affirming that god
will answer for his lifes partner to be on my desk
yesterday next i thought for a few seconds says it all
and take into account the significance of the story was
all she just wanted to show you this last tuesday
```

8 Text Generation With Markov Chains

8.1 Metadata

The Sherlock Holmes Stories dataset is published by Devji Chhanga in Kaggle. It is an open-source dataset (license: CC0: Public Domain) and contains 67 Sherlock Holmes short stories written by Arthur Conan Doyle.

8.2 Implementation

Sherlock Holmes stories written by Arthur Conan Doyle, have remained very popular over the years. 67 Sherlock Holmes short stories are used for this implementation. After installing and importing Natural Language Toolkit and needed Python libraries, reading and text cleaning process is performed on each story. As next step the Markov function is implemented including the calculation of transition probabilities for 208714 states in this case. Finally the cleaned text is fitted to the model in order to generate sentences and paragraphs. A reading sample of some generated sentences are shown below. For more details and complete results please check the project GitHub repository.

Generated sentences beginning with "dear holmes" :

```
dear holmes i exclaimed so far i was newly come
dear holmes you are going on the moor at night
dear holmes i fear that unless you can lay your
dear holmes am i wrong yes that is the question
dear holmes what do you think he is the central
```

Reading sample of the generated Sherlock Holmes story:

```
the case fortune has been your friend this was
not a pleasant walk of the weald station i never
needed it more puzzled than ever i listened to the
dispute which was examined within a few weeks
afterwards i learned in business so it had if anything
augmented it the door was opened and took out a note
upon her face illuminated by the collar and cuffs this
he buttoned tightly up in spite of her dress a small
packet of technical papers holmes thrust the chunks of
wood into the chink until at last when he saw your
theft but could find no case which may account for it
```

9 Conclusion

This project was absolute fun! The powerful but simple Markov chains can create a predictive text model. It models the transition probability between states, where in NLP each state is represented by terms/words. The generated output is relatively good but it is not good as human written text. The reason is that Markov Chains just use the transition probabilities. Combining of a RNN model (LSTM architecture) with Markov Chain delivers better results, compared to other applications which use solely Markov Chains method. Regardless, Markov Chains can be used as a predictive text application, for instance assisting in google search or writting a sms.
One of the faced challenges in this project was regarding hardware and computational power. The book generator application crashes multiple times because of memory requirements. Using Google Colaboratory service, solved the problem regrading RAM.

References

[1] P. A. Gagniuc, *Markov Chains: From Theory to Implementation and Experimentation*. Wiley, 2017.

[2] P. Potash, A. Romanov, and A. Rumshisky, "Ghostwriter: Using an lstm for automatic rap lyric generation," no. 1, 2015.

[3] P. Potash and A. Romanov, "Evaluating creative language generation: The case of rap lyric ghostwriting," no. 1, 2016.